# METHOD AND PROTOCOL FOR MANAGING BROADBAND IP SERVICES IN A LAYER TWO BROADCAST NETWORK

## Field of the Invention:

The present invention relates to a load balancing fault tolerant scalable IP broadband

5    service management protocol that relates to flow oriented switching and, more particularly, to a method and protocol for managing large number servers that provide broadband IP services in a layer 2 broadcast network by advertising and directing service from intermediate system to any available end system, and also advertise and direct packet flow between intermediate system and end system based on flow requirement.

10

## Background of the Invention:

The current way of communication between intermediate system and end system is relying on ARP (address resolution protocol). The ARP sends out a broadcast to all end systems to look for a layer two address of an IP address. This mapping from one IP address

15    to only one MAC address just provides reachable information. This invention will provide more than just reachable information to allow more intelligent forwarding decision to satisfy the service requirement that includes the traffic quality and other policies.

In order to scale the service to large number of physical servers (end systems), it would be necessary to have a method and protocol to improve the current way of communication

20    between intermediate system and end system (server) that is using ARP protocol, this invention allows one IP address to map to multiple physical servers identified by MAC address, and to provide intelligent selection and forward algorithm to distribute application flow among all the end systems (servers).

25    ## Summary of the Invention:

This invention basically will not use the ARP that sends out a broadcast to all end systems to look for a layer two address of an IP address. Instead, the intermediate system will listen to the service information advertised by the servers and relayed by supervisor system and store them into its server table and forward the newly requested traffic to a server that is

30    looked up from this server table. The selection of the target server depends on service

congestion advertisement that will advertise the service availability and response time to help the intermediate system to select the less congested server to send the request to. After server accepts the service request, it will advertise the flow routing information. The flow routing information will specify usually itself as the destination (using layer two Ethernet MAC address) for the traffic flow that it is serving. Not just destination address but also the source address and other protocol number in the packet header could be specified as the traffic flow matching criteria. These flow advertisements will be stored in the flow table in the intermediate system in addition to server table.

After the flow is installed by the intermediate system, the later packets that match the flow will be forwarded to the end system that is based on the result of looking up the flow table. The result will indicate which server (represented by a layer two MAC address) to send to for this flow. The flow table will be always looked up first, if the intermediate system could not find a match. The packet will be treated as a new first service request packet that is looking for service. The intermediate will then look up the server table to find one available server to send the request to. Service can be withdrawn and added on the fly, and flow can be terminated through advertisement after it is finished, time out or other policies, but these will be transparent to the end users. All these servers are all sharing the same IP address that allow the client to access like one big server. Any of the protocol number and any other number pool could be centrally controlled by SS (supervisor system) to guarantee the uniqueness of the traffic flow and proper number resource management. The invention features the load balancing with fault tolerance with Service Information and Flow Advertisement Protocols among the intermediate systems and end systems.

The invention also features the service type and flow advertised in a pattern-matching format, which allow intermediate switching system to classify the packet without knowing the application details.

The invention also features the service information advertised by the end system (server) that contains the service attributes of service congestion, capability and others to help the server management in this layer two network.

The invention also features the flow information advertised by the end system (server) that contains the flow attributes of quality of service requirements as a label distribution

protocol in a layer two network.

The invention also features the method of allowing multiple physical servers sharing a single IP address in a way that is transparent to the client, through the use of Assigned Numbers Authority protocol with Service Information and Flow advertisement protocol.

5 The invention also features the use of label (label number is centrally controlled through Assigned Numbers Authority protocol) to differentiate between the packets that are using the none-unique overlap private IP address by adopting the uniquely assigned source label.

This invention will improve the current way of communication between intermediate system and server using ARP protocol. The invention will allow one IP address to map to 10 multiple physical servers identified by layer two address and still can provide uniqueness of communication between each physical server and client.

The system of the present invention includes intermediate system, end system, and supervisor system. The IS (intermediate system) is IP router or switch device that receives traffic from client and forward it to the end system on this layer 2 network and on the other 15 end it also forwards the response traffic from the end system to the client. The ES (end system) could be any kind of servers including HTTP server, FTP server, firewall proxy server, IPSEC tunneling server, Streaming media server, content cache server and NAT (Network Address Translation) server etc. The SS (supervisor system) is a special system that handles the service registration from all the systems on this layer two network. It also 20 replies the request of any new system that looks up the server list for a particular service on this network. It could also runs as an "Assigned Numbers Authority" server for this layer 2 network and also runs as a management agent that manages all the systems registered to it. Since all the servers could share one IP address, there is a need for centralized controlled to guarantee the uniqueness of the traffic flow classification between the client and the 25 physical server. Assigned Numbers Authority Protocol achieves this. A back up supervisor system can be provided to support redundancy if the primary supervisor system failed. This layer two network could be built by layer two switching devices (like Ethernet switches) or physical repeaters (like Ethernet hubs).

30 **Brief Description of the Drawings:**

- 3 -

FIG. 1 is a block diagram illustrating a system of the present invention having intermediate system, end system, and supervisor system.

FIG. 2(a) is a sequence diagram illustrating service registration sequence of service information protocol of the present invention.

5    FIG. 2(b) is a sequence diagram illustrating service de-registration sequence of service information protocol of the present invention.

FIGS. 3(A), 3(B) AND 3(C)

FIG. 3(a) and 3(b) illustrate service addition and deletion update sequence of service information protocol of the present invention.

FIG. 4(a) is a sequence diagram illustrating server congestion multicast message of 10    service control advertisement of the present invention.

FIG. 4(b) is a sequence diagram illustrating the triggered server congestion unicast message of service control advertisement of the present invention.

FIG. 4(c) is a sequence diagram illustrating redirect message sequence of service control advertisement of the present invention.

15    FIG. 5 is a diagram illustrating flow advertisement sequence of the present invention.

FIG. 6 is a diagram illustrating assigned number authority protocol sequence of the present invention.

FIG. 7 is the new back up supervisor system to retrieve the assigned number list from primary supervisor system.

20    FIG. 8 is a diagram illustrating label encapsulation in user packet.

FIG. 9 is a diagram illustrating message format with or without VLAN priority.

FIG. 10 is a diagram illustrating protocol message format.

FIG. 11 is a diagram illustrating message format for common message header.

FIG. 12 is a diagram illustrating message format for service registration request.

25    FIG. 13 is a diagram illustrating message format for services de-registration request.

FIG. 14 is a diagram illustrating message format for service registration acknowledgement.

FIG. 15 is a diagram illustrating message format for service de-registration acknowledgement.

30    FIG. 16 is a diagram illustrating message format for service type matching rule.

FIG. 17 is a diagram illustrating message format for service update request.

FIG. 18 is a diagram illustrating message format for service deletion update.

FIG. 19 is a diagram illustrating message format for service delete update.

FIG. 20 is a diagram illustrating message format for service update acknowledgement.

5    FIG. 21 is a diagram illustrating message format for service control advertisement.

FIG. 22 is a diagram illustrating message format for flow advertisement.

FIG. 23 is a diagram illustrating message format for flow advertisement acknowledgement.

FIG. 24 is a diagram illustrating message format for flow attributes.

10    FIG. 25 is a diagram illustrating message format for Assigned Number Request.

FIG. 26 is a diagram illustrating message format for Assigned Number Acknowledgement.

FIG. 27 is a diagram illustrating message format for Assigned Number Update Request.

FIG. 28 is a diagram illustrating message format for Assigned Number Update.

15    FIG. 29 is a diagram illustrating message format for Assigned Number Update Acknowledgement.


## Detailed Description of the Invention:

As shown in FIG. 1, the systems of the present invention include intermediate system,
20    end system, and supervisor system. The IS (intermediate system) is IP router or switch kind of device that receives traffic from outside, and forward it to the end system. On the other end, it also forwards the response traffic from the end system to the outside. The ES (end system) could be any kind of servers including HTTP server, FTP server, firewall proxy server, IPSEC tunneling server and NAT (Network Address Translation) server etc. The SS
25    (supervisor system) is a special system that handles the registration from all the systems on this layer two network. It also respond to the requests from any new system that looks up the server list for a particular service on this network. It also runs as an "Assigned Numbers Authority" server for this layer 2 network. It could also runs as a management agent that manages all the systems registered to it. Since all the servers are sharing one IP address,

there is a need for centralized controlled to guarantee the uniqueness of the traffic flow between the client and the physical server. A back up supervisor system can be provided to support redundancy if the primary supervisor system failed. This layer two network could be built by layer two switching devices (like Ethernet switches) or physical repeaters (like

5    Ethernet hubs).

A service is usually provided by multiple end systems (servers), an intermediate system will learn these services advertised by the end systems and directs the traffic based on the service information advertisements. If a system is providing proxy service like firewall application proxy, it is both an intermediate system and end system. It acts like a server

10   when it accepts requests from the client and also acts like client when it initiates a request to the application server that actually provides the service.

In another aspect, the intermediate system is providing the forwarding service based on flow, service type or destination address.

A service may include both processing and forwarding. A system will try to process the

15   packet first, if it cannot process it, it will try to forward it based on flow, if flow information is not available, then it will look up the service type. The last resort is to forward the packet based only on the destination address. But the processing or forwarding decision also includes also the "drop packet" decision.

20   # <u>Service</u>

A service is described by pattern matching rule, which allows the intermediate system that parses the packet based on these rules to determine a match. Hence the intermediate system does not have to be "application aware" as long as it can execute the pattern matching rule that usually can be done by very fast network processor. The fields to be

25   parsed will sometimes need to include not just the destination, but also the source information for a service. One example is the address translation service that will check both the source address and destination address to determine whether address translation is needed. A more precise definition of service will be defined in the description of message format.

In multicast addresses situations, we use AllSupervisors to represent a well-known layer two multicast address for both the primary supervisor system and back up supervisor system, AllServerOfServiceX to represent a layer two multicast address for all the servers that provide the service X, and AllClientOfServiceX to represent a layer two multicast address

5    for all the clients that are interested in the service X.

As shown in FIG. 2(a), a sequence diagram illustrates service registration message sequence of service information protocol of the present invention. When an end-system (server) just comes up, it will send this multicast message to AllSupervisors about the service it provides with its own layer two address and keep-alive interval. The service is

10    specified as pattern matching rules. The (server) end-system should re-transmit until the supervisor system acknowledges its service registration. The acknowledgement by the supervisor system will assign a multicast address to the servers that provide this type of service (i.e. AllServerOfServiceX) as well as the multicast address of all the clients that are interested in this type of service (i.e. AllClientOfServiceX). The supervisor system could

15    also overwrite the keep-alive interval if necessary. The server should acknowledge the supervisor system by sending its first keep-alive message and so on.

The service de-registration message allows (server) end-system to withdraw the service it provides. Of course, the (server) end-system also needs to re-transmit until the supervisor system acknowledges its service de-registration.

20    In FIG. 3(a) and (b), if there is any server list change, the supervisor system should send this update multicast message of server addition and deletion list to AllClientOfServiceX. The server change list must fit into one packet to allow each client to acknowledge the update. If there is more than one packet which is split into multiple update messages and waits for all the acknowledgements to come back before the next one is sent.

25    In FIG. 3(c), a new (client) intermediate system (possibly an end system too) can send this multicast message to AllSupervisors to ask for the server list for a specific type of service and indicate its own keep-alive interval. The supervisor system should respond with a server update message with the server list for that specific type of service, which was registered to it. If the server list is too long to fit into one packet, the response packet should

30    assign a sequence number and a two bit flag in each packet to indicate it is the init (first)

and more packets. Each packet should be acknowledged by the new intermediate system based on the sequence number before the supervisor system can send the next one. The reply from the supervisor system also assigns a multicast address to the clients that are interested in this type of service (i.e. AllClientOfServiceX) as well as the multicast address

5    of all the servers that provide this type of service (i.e. AllServerOfServiceX). The supervisor system could also overwrite the keep-alive interval if necessary. After the last acknowledgement for the server update message, the server should wait no more than one keep-alive-interval to send out its first keep-alive message.

FIG. 4(a) is a sequence diagram illustrating congestion message sequence of service

10   control advertisement of the present invention. When a server is congested and drops the request packet because it cannot take more requests for the service it provides, it should send this multicast message to AllSupervisors and AllClientOfServiceX about the congested service X provided by it. The congestion message will indicate the 'temporary out of service time'. When the other systems received such a congestion message, they should

15   flag the congested server as inactive in its server list for that particular service until the 'out of service time' status expires.

When a server is in 'temporary out of service' state for new request, and it receives a new request for that service, it must respond a unicast service congestion advertisement message to the requesting node. This is illustrated in FIG 4(b). It optionally may also re-

20   direct (forward as is) the request to other server who can also provide the same service.

FIG. 4(c) is a sequence diagram illustrating redirect message sequence of service control advertisement of the present invention. When a server tries to respond to the request, it picks up one router to send to from its (routing) server list, it may not choose the best next hop router, but this message (works similar to an ICMP message) can redirect the server to

25   pick up a better router. A server could also use this message to redirect the user request to other better server node based on the service configuration or other requirement.


## Flow

A flow is a description of a traffic flow from source to destination. It could be same as a

TCP connection that specifies the source IP address, destination IP address, source TCP port and destination port, but it also can be specified just by source IP address and destination IP address and destination TCP port without source TCP port. The latter case is particularly important to maintain a persistent connection between the client and server.

5   The specification of a flow is totally determined by the server, i.e. dependent server application requirement. In this invention, the server that accepts the connection request will advertise the flow information to intermediate system to instruct the intermediate system how to forward later packets that match the flow specification. A flow could be established from end system to intermediate system, and can also be extended and established from the

10   intermediate system further to its next hop upstream intermediate system neighbor. A flow is also described by pattern matching rule, which allows the intermediate system that parses the packet based on these rules to determine a match. Hence the intermediate system does not have to be "application flow aware" as long as it can execute the pattern matching rule that usually can be done by very fast network processor. A flow is unidirectional.

15   The flow from intermediate system to server is advertised by server, with the flow-matching rule that is identified by a flow label assigned by the advertising server.
The flow from server to intermediate system is still advertised by server, but the flow label is assigned by the intermediate system in the flow acknowledgement packet.
A more precise definition of flow will be defined in the description of message format.

20   FIG. 8 represents label encapsulation in user packet of the present invention. The user traffic in this layer two network will be inserted with four-byte value with label information. If the packet matches a flow, the flow label will be inserted into a normal Ethernet frame by either intermediate system or a server. If the packet doesn't match any flow yet, the four-byte value will be filled with a source label that is used to indicate the source of the

25   packet. Two new frame types will be needed to uniquely determine the frame format. One is for flow label frame; the other is for source label frame.
Optionally, the priority only VLAN tag frame might be inserted before the label.
The frame formats with or without VLAN priority are shown in FIG. 9. The label can also be used to differentiate the private address. One example of this source label is to embed

30   the VPN identification or the incoming router device ID and interface number. For the

packet using none-unique private IP address, the source label can be used to uniquely classify a flow. The VPN identification could be configured to associate to an interface of a system. Different VPN identification number in the packet can differentiate packets (from different interfaces) that are using the same source private IP address to a the same desti-

5    nation IP address.

In flow advertisement protocol, the (server) end-system should send this multicast message to AllClientOfServiceX about the flow with attributes and actions it accepted with its own layer two address. The flow is specified in pattern matching rules. The (server) end-system should re-transmit until original requesting intermediate system acknowledges

10   its flow advertisement. The flow attributes will be specified in the flow advertisement. A static user configured flow is also possible to allow the flow based policy routing.

FIG. 5 is a diagram illustrating advertisement sequence of the present invention. The flow advertised by the end system (server) would multicast to AllClientOfServiceX. But the advertised message will indicate the original requesting intermediate system. Those

15   intermediate systems are not the original requesting intermediate system, which may store this flow information into their flow table or process this flow information based on their own implementation and policies. This advertisement does not require acknowledgement from each of the interested clients except the original requesting intermediate system to avoid too much flow acknowledgement traffic.

20

## Assigned Numbers

For assigned numbers, within this layer two network, there are numerous parameters, such as IP addresses, under same IP address the TCP or UDP port number, certain fields in layer three to layer seven header or content and many others need to be controlled and

25   managed. Some of them are required to guarantee the uniqueness of a traffic flow; others are common resources that needs be centrally controlled. The sharing of one IP address for many servers that provide the same service requires that the values used in certain parameter fields in the packet header and content to be assigned uniquely. It is the task of the Assigned Numbers Authority server to make those unique number assignments as re-

30   quested and also maintain a registry of the currently Assigned values.

In assigned numbers authority protocol, the end system or intermediate send this multicast message to AllSupervisors to request to lease one or a range of certain parameters to be used. The supervisor system replies and grants the range of the number from its pool for that specific parameter with the lease time equal or smaller than the requested least time.

5 When the requesting system is detected to be dead by supervisor system, those number granted to it will be withdrawn back to the pool by the supervisor system.

### Synchronization of the registry of the currently assigned and unassigned numbers

Also, to synchronize the registry of the currently assigned and unassigned values be-

10 tween the backup supervisor system and primary supervisor system, the back up supervisor system can send a unicast request to primary supervisor system about the assigned and unassigned numbers for a specific number type. The primary supervisor system should response with a number update message with the assigned and unassigned numbers for that specific type of number. If the type of number is too many to fit in one packet, the response

15 packet should assign a sequence number and a flag in each packet to indicate it's the first, middle or last packet. Each packet should be acknowledged by the new back up supervisor system based on the sequence number before the primary supervisor system can send next one.

20 ### Supervisor and back up supervisor system selection process

In back up supervisor system redundancy support, when a supervisor system (whether back up or not) comes up, it always tries to inquire the back up supervisor system from the existing supervisor system by sending a multicast service update request message to AllSupervisors. If there is any existing primary supervisor system that replies this service update

25 request without a back up supervisor system available, then the new supervisor system will become a back up supervisor system, and it will synchronize the server table with primary supervisor system through Server Update Message Protocol. Also, the back up supervisor system will synchronize numbers that are assigned and unassigned by the primary supervisor system through Number Update Message protocol. After the last number update

30 message was acknowledged, the back up supervisor system will periodically send inquiry

unicast request for primary supervisor system and listen all the messages sent to the All-Supervisors multicast address. If there exists a back up supervisor system, it will periodically send out the inquiry to try to become a back up supervisor system until the existing back up supervisor system actually disappeared. If there is no response for the service

5 update request for an (dead) interval (3 times of service inquire intervals), then the new supervisor system will become the primary supervisor system. (But there is no back up supervisor system.) If the new supervisor system received an inquiry request about the back up supervisor system to primary supervisor system during the (dead) interval with the MAC address of received message is lower than its own MAC address, then the new su-

10 pervisor system itself will still become primary supervisor system at the end of (dead) interval. If the new supervisor system received an inquiry request to primary supervisor system message during the (dead) interval with the MAC address of received message is higher than its own MAC address, then the system with highest MAC address will be the primary supervisor system and the second highest MAC address will become the back up

15 supervisor system. After the primary supervisor system is found or elected, the back up supervisor system will periodically send inquiry request for primary supervisor system, if primary supervisor system received the request, it will refresh back up supervisor system age timer and replies the request and so on. If the primary supervisor system doesn't receive the back up request at end of age time, it will remove the back up supervisor system and

20 may send a SNMP trap. On the other hand, if there is no response for an (dead) interval (3 times of service inquire intervals), then the back up supervisor system will become the primary supervisor system.

**Protocol message format**

25 FIG. 10 is a diagram illustrating protocol message format of the present invention. The packet could be carried by Ethernet frame or carried by IP or UDP.

| DMAC | SMAC | Type | Message |
|------|------|------|---------|

We will find a unique type for the frame type.

| IP header | Message |
| --- | --- |

We will find a unique protocol type in IP header.

| UDP header | Message |
| --- | --- |

We will find a unique port number in UDP header.

5

| 2 byte Message Type | 2 byte Message length | Message |
| --- | --- | --- |

FIG. 11 is a diagram illustrating common message header format of the present invention.

The message type will have

10

1000: Service Registration Request

1001: Service Registration Acknowledgement

1002: Service De-registration Request

1003: Service De-registration Acknowledgement

15

1100: Service Update Request

1101: Service Add Update

1102: Service Add Update Acknowledgement

1103: Service Delete Update

20   1104: Service Delete Update Acknowledgement

1200: Service Control Advertisement

1300: Flow Advertisement

25   1301: Flow Advertisement Acknowledgement

1400: Assigned Number Request

1401: Assigned Number Acknowledgement


5     1500: Assigned Number Update Request

1501: Assigned Number Update

1502: Assigned Number Update Acknowledgement


For the keep-alive message, a system periodically sends Service Control Advertisement multicast mes-

10    sage to AllClientOfServiceX with keep-alive interval service attribute to refresh its age time in supervisor

system (also a client of service X) and all the clients of the service X. The current response time will also be

attached as the metric as the loading factor to let client (IS) select the less congested server.

| 4 byte Service Type | Server Address | 4 byte Number of Service attributes | Service attribute | ... |
|---|---|---|---|---|

15

| 2 byte Service attribute type | 2 byte Service attribute length | Service attribute value |
|---|---|---|

Server address is 8 byte field, the first two bytes determine it's MAC address or

IP address. 00 00 is MAC, 00 01 is IP.


20    Keep-alive interval and Server current Response time are two of the service attribute types,

1 for Keep-alive interval, 2 Server current Response time


The keep-alive interval is the time interval that the system will send the periodical

keep-alive message.

25    Server current Response time is the current time difference between the request packets (ex. HTTP request)

entry time stamp by the Ethernet driver and the time stamp the response packet sent to the wire. The current

time difference should be recorded by sending drivers and averaged within the keep-alive interval.

Supervisor system may calculate the average response time based on each server's latest current response time within the interval that the supervisor system sent to its parent supervisor system.

FIG. 12 is a diagram illustrating message format for service registration request of the

5    present invention.

| Server Address | Service type matching rule | 4 byte Number of Service attributes | Service attributes (keep-alive etc.) | Service type matching rule | ... |
|---|---|---|---|---|---|

Service type matching rule is variable length that is defined in later section.

10   Server address is 8 byte field, the first two bytes determine it's MAC address or

IP address. 00 00 is MAC, 00 01 is IP.

FIG. 13 is the message format for service de-registration request, the format is as following:

| Server Address | 4 byte Service type | 4 byte Service type | 4 byte Service type | ... |
|---|---|---|---|---|

15

FIG. 14 is a diagram illustrating message format for service registration acknowledgement of the present invention.

| 4 byte Service type | Group Addresses | 4 byte precedence | 4 byte Service type | Group Addresses | ... |
|---|---|---|---|---|---|

20

| Server Unicast Address | Server Multicast Address | Client Multicast Address |
|---|---|---|

Server address is 8 byte field, the first two bytes determine it's MAC address or

IP address. 00 00 is MAC, 00 01 is IP unicast or multicast.

The server unicast address is the server address that registers the service. The server multicast address is the multicast address for all the servers that are serving the registered

5   service assigned by supervisor system. The client multicast address is the multicast address for all the clients that are interested in the registered service also assigned by supervisor system. Precedence is used for service matching tie breaking if there are two or more matching service rules. The intermediate system's destination only forwarding service should have lower precedence than the others. This means the searching the routing table

10   for a next hop node has lower priority than searching the flow table and (processing) server table. FIG. 15 is the message format for service de-registration acknowledge, the format is as following:

| 4 byte Service type | Server Address | 4 byte Service type | Server Address | ... |
|---|---|---|---|---|

15   FIG. 16 is a diagram illustrating message format for service type matching rule of the present invention. In service classification, the format to describe a service type in pattern matching way is defined in figure 11. A service type is uniquely classified by a packet-parsing rule. If somehow there are multiple packet-parsing rules for the same service type, it is still classified as a different service type although the server that provides the

20   service will serve them all. The parsing rule for a specific service type consists of a serial of pattern matching fields. The related fields especially in the same layer's header can be grouped together as a sub-type. The most significant bit in the service type or sub-type will indicate if this is the final field or an intermediate sub-type or type. The bit location is starting from the beginning of the field's associated layer header and the hierarchy could go

25   all the way up to layer 7. The FIG 16 message format for service type matching rule is as following:

| 2 byte Service type | 2 byte Service length | Variable length service type matching rule |
|---|---|---|

| 2 byte Sub-type | 2 byte Sub-type length | Variable length sub-type matching rule | 2 byte Sub-type | 2 byte Sub-type length | Variable length sub-type parsing rule |
|---|---|---|---|---|---|

5

| 6 byte Field type | 2 byte Field length | Variable length Field value | 6 byte Field type | 2 byte Field length | Variable length Field value | 6 byte Field type | ... |
|---|---|---|---|---|---|---|---|

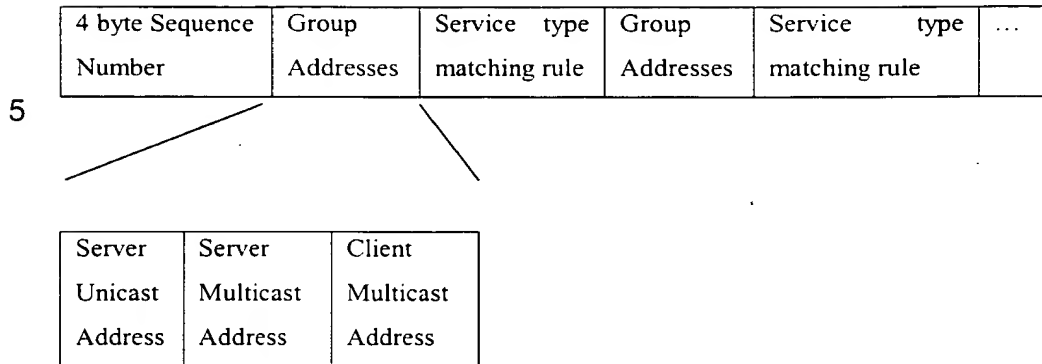| 2 byte Starting bit location | 2 byte bit length | 2 byte operator |
|---|---|---|

10

Note that the unit of length is in number of bytes. The service type-matching rule consists of multiple level and multiple fields to be simultaneously matched. These pattern

15 matching can be executed much faster by the pattern matching network processor than the normal procedure oriented CPU. The starting bit location is the offset from the beginning of the packet header that is deduced by the type or sub-type. The operator consists of ">", " <", ">=", "<=", "==" and "!=". The unit of length of the field value is counted in bytes, the field value could have unused leading zero bits. Same starting bit location and bit length

20 can appear more than once if the operator is different.

In the FIG 17 message format for service update request, as shown in the following:

| 4 byte Service type | 4 byte Service type | 4 byte Service type | 4 byte Service type | ... |
|---|---|---|---|---|

The replied message will be the services add update message.

FIG. 18 is a diagram illustrating message format for service addition update of the present invention.

| 4 byte Sequence Number | Group Addresses | Service type matching rule | Group Addresses | Service type matching rule | ... |
|---|---|---|---|---|---|

5

| Server Unicast Address | Server Multicast Address | Client Multicast Address |
|---|---|---|

The highest 2 bits of sequence number will indicate the first, middle or last update packet.

10   Bit 31, 30 of sequence number is init and more bits.

| Init | more | |
|---|---|---|
| 1 | 0 | single packet |
| 1 | 1 | first |
| 0 | 1 | middle |
| 0 | 0 | last |

15

FIG. 19 is a diagram illustrating message format for service delete update of the present invention.

| 4 byte Sequence Number | Group Addresses | 4 byte Service type | Group Addresses | 4 byte Service type | ... |
|---|---|---|---|---|---|

20

| Server Unicast Address | Server Multicast Address | Client Multicast Address |
|---|---|---|

The highest 2 bits of sequence number will indicate the first, middle or last update packet.

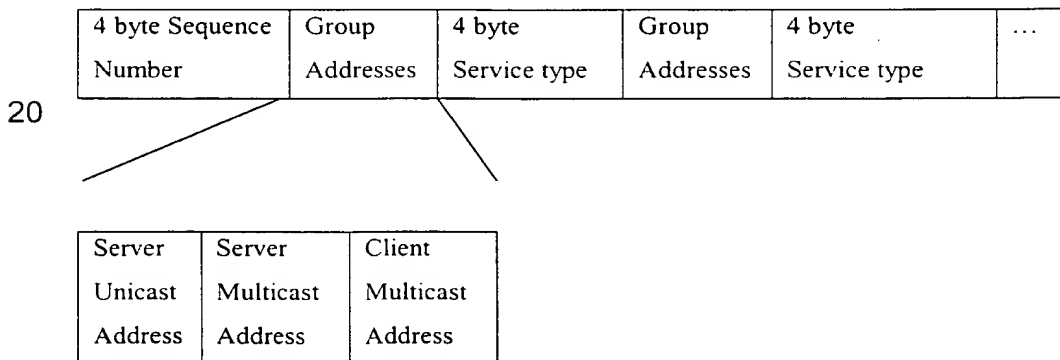Bit 31, 30 of sequence number is init and more bits.

Init    more

1       0       single packet

5       1       1       first

0       1       middle

0       0       last

FIG. 20 is a diagram illustrating message format for service updates acknowledge of the

10      present invention. This is the acknowledgement for both add or delete service update.

| 4 byte Sequence Number | Group Addresses | 4 byte Service type | Group Addresses | 4 byte Service type | ... |
|---|---|---|---|---|---|

| Server Unicast Address | Server Multicast Address | Client Multicast Address |
|---|---|---|

15

For deletion of a service, the server and client addresses should be filled in with zero.

The highest 2 bits of sequence number will indicate the first, middle or last update packet.

Bit 31, 30 of sequence number is init and more bits.

Init    more

20      1       0       single packet

1       1       first

0       1       middle

0       0       last

25      FIG. 21 is a diagram illustrating message format for service control advertisement of the

present invention.

| 4 byte Service Type | Server Address | 4 byte Number of Service attributes | 2 byte Service attribute type | 2 byte Service attribute length | Attribute value | 2 byte Service attribute type | ... |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

Two more service attribute types for service control message,

3 for "temporary out of service" by a congested system;

4 for "redirect" the packet to other better system by intermediate system or server.

5

| Temporary Out of service | Length | Out of service time |
|---|---|---|
| | | |

| Redirect | Length | Server addresses |
|---|---|---|
| | | |

Length will determine the length of all server addresses.

Server addresses are all the possible servers that can be redirected.

10

FIG. 22 is a diagram illustrating message format for flow advertisement of the present invention.

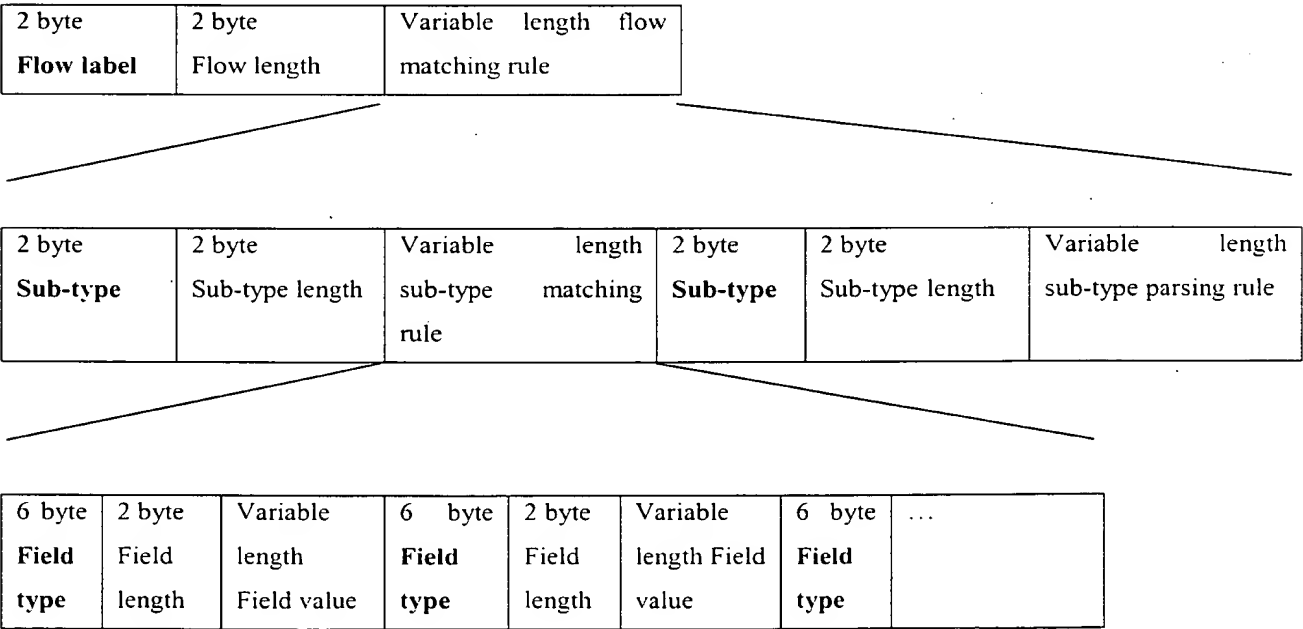| Flow matching rule | 4 byte Number of flow attributes | Flow Attributes | Flow matching rule | ... |
|---|---|---|---|---|
| | | | | |

15    FIG. 23 is a diagram illustrating message format for flow advertisement acknowledgement of the present invention.

| Flow label | 4 byte Number of flow attributes | Flow Attributes | Flow label | 4 byte Number of flow attributes | Flow Attributes | Flow label | ... |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

20    The flow label is 4 bytes. The intermediate system could acknowledge the flow with a flow attribute to overwrite a new flow label. One of the flow attributes is action attribute;

one of the action attribute values is to withdraw the flow. The intermediate system could acknowledge the flow with a flow attribute to overwrite a new flow label.

The format to describe a flow is defined in figure 18. A flow is uniquely classified by a packet-parsing rule and will be identified by a 32-bit label. The range of the label assigned to a server is going through the Assigned Number Authority Protocol. This makes the flow label unique across the whole layer two network. If somehow there are multiple packet-parsing rules for the same flow, it is still classified as a different flow although the intermediate system that directs them all to the same end system or possible next intermediate system. The parsing rule for a specific flow consists of a serial of pattern matching fields. The related fields especially in the same layer's header can be grouped together as a sub-type. The most significant bit in the service type or sub-type will indicate if this is the final field or an intermediate sub-type or type. The bit location is starting from the beginning of the field's associated layer header and the hierarchy could go all the way up to layer 7.

| 2 byte **Flow label** | 2 byte Flow length | Variable length flow matching rule |
| --- | --- | --- |

| 2 byte **Sub-type** | 2 byte Sub-type length | Variable length sub-type matching rule | 2 byte **Sub-type** | 2 byte Sub-type length | Variable length sub-type parsing rule |
| --- | --- | --- | --- | --- | --- |

| 6 byte **Field type** | 2 byte Field length | Variable length Field value | 6 byte **Field type** | 2 byte Field length | Variable length Field value | 6 byte **Field type** | ... |
| --- | --- | --- | --- | --- | --- | --- | --- |

| 2 byte Starting bit location | 2 byte bit length | 2 byte operator |
|---|---|---|

Note that the unit of length is in number of bytes.

The flow-matching rule consists of multiple levels and multiple fields to be simultaneously matched. These pattern matching can also be executed much faster by the pattern matching network processor than the normal procedure oriented CPU. The starting bit location is the offset from the beginning of the packet header that is deduced by the type or sub-type. The operator consists of ">", " <", ">=", "<=", "==" and "!=".

The unit of length of the field value is counted in bytes. the field value could have unused leading zero bits. Same starting bit location and bit length can appear more than once if the operator is different. In flow attribute, flow could have attributes of traffic engineering requirements, quality, priority, label of the MPLS and other flow oriented information.

FIG 24 Message format for flow attribute

| 2 byte flow Attribute type | 2 byte Attribute length | Variable length Attribute value |
|---|---|---|

The most significant bit in the Attribute type will indicate if this is the last attribute.

Flow attribute - action

The attribute type 1 is action attribute.

The flow-action attribute consists of "permit", "deny", "MPLS label insertion", "MPLS label removal", "flow withdraw" and any other policy configuration matching action items.

Flow attribute - label value

The attribute type 2 is label value attribute.

This could be used by intermediate system to overwrite the label value for an incoming flow advertised by server.

Flow attribute - designated system

The attribute type 3 is to identify the designated system that should acknowledge this flow advertisement.

5    Flow attribute - bandwidth and maximum delay

The attribute type 4 is to specify the bandwidth requirement and maximum delay of this flow.

Zero is reserved for best effort traffic.

Flow attribute - DiffServ and TOS (Type Of Service) modification

10   The attribute type 5 is to specify the new value for the TOS or DiffServ in the IP packet header.

FIG 25 Message format for Assigned Number Request

| 4 byte total number of number types | 2 byte Num-ber type | 2 byte Length | Values | 2 byte Number type | ... |
|---|---|---|---|---|---|

15

Values contain the starting number and the length of the number requested etc.

FIG 26 Message format for Assigned Number Acknowledge

| 4 byte total number of number types | 2 byte Num-ber type | 2 byte Length | Values | 2 byte Number type | ... |
|---|---|---|---|---|---|

20

Values contain the starting number and the length of the number granted etc.

FIG 27 Message format for Assigned Number Update Request

| 4 byte total number of number types | 2 byte Number type | 2 byte Number type | 2 byte Number type | 2 byte Number type | ... |
|---|---|---|---|---|---|
| | | | | | |

FIG 28 Message format for Assigned Number Update

5

| 4 byte Sequence number | 4 byte total number of number types | 2 byte Number type | 2 byte Length | Values | 2 byte Number type | 2 byte Length | ... |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

The highest 2 bits of sequence number will indicate the first, middle or last update packet.

Bit 31, 30 of sequence number is init and more bits.

Init   more

10   1   0   single packet

1   1   first

0   1   middle

0   0   last

15   FIG 29 Message format for Assigned Number Update Acknowledge

| 4 byte Sequence number | 4 byte total number of number types | 2 byte Number type | 2 byte Length | Values | 2 byte Number type | 2 byte Length | ... |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

The highest 2 bits of sequence number will indicate the first, middle or last update packet.

Bit 31, 30 of sequence number is init and more bits.

20   Init   more

1   0   single packet

| 1 | 1 | first |
| 0 | 1 | middle |
| 0 | 0 | last |

The present invention has been described hitherto with exemplary preferred embodiments. However, it is to be understood that the scope of the present invention need not be limited to the disclosed preferred embodiments. On the contrary, it is intended to cover various modifications and similar arrangements with the scope defined in the following appended claims. The scope of the claims should be accorded the broadest interpretation so as to encompass all such modifications and similar arrangements.